AMENDMENTS TO THE CLAIMS

1.-27. (Canceled)

28. (Currently amended) A method of providing services of an application comprising: providing a plurality of network interfaces;

providing a plurality of CPU's;

running ~~an~~ a separate instance of the application for each one of the plurality of network interfaces;

designating a separate one of said plurality of CPU's to each separate instance and in turn to the network interface running its respective separate instance; ~~and~~

binding ~~a separate~~ each one of said plurality of network interfaces to ~~each~~ only the CPU designated to the separate instance of the respective network interface, whereby each network interface and its separate instance is handled solely by the CPU to which that network interface is bound;

providing a processing queue for each of the plurality of CPU's; and

assigning a separate one of the processing queues to each one of the plurality of CPUs, wherein the processing queue assigned to a particular CPU provides single threaded processing of data related to an instance of the application to which the CPU is bound.

29. (Currently amended) ~~A~~ The method as recited in claim 28, further comprising: assigning a separate network address to each one of the plurality of network interfaces.

30. (Currently amended) ~~A~~ The method as recited in claim 29, wherein each separate network address is an Internet Protocol (IP) address.

31. (Currently amended) ~~A~~ The method as recited in claim 29, wherein said step of running an instance of the application for each one of the plurality of network interfaces comprises:

for each one of the plurality of network interfaces, initiating a listener that listens for the network address that is assigned to that network interface.

32. Canceled

33. (Currently amended) ~~A~~ The method as recited in claim ~~32~~ 28 wherein each processing queue is a sequential queue (s-queue).

34. (Currently amended) ~~A~~ The method as recited in claim ~~32~~ 28, wherein each single threaded processing is uninterrupted while processing the data related to an instance of the application.

35. (Currently amended) ~~A~~ The method as recited in claim 33, further comprising:

receiving data packets;

processing each data packet to determine a particular one of the processing queues corresponding to connection classifier information in the data packet; and

routing the data packet to the determined processing queue.

36. (Currently amended) ~~A~~ The method as recited in claim 35, further comprising: processing the packet by the determined processing queue.

37. (Currently amended) ~~A~~ The method as recited in claim 36, further comprising:

if the determined processing queue is busy, waiting before the step of processing the data packet by the determined processing queue.

38. (Currently amended) ~~A~~ The method as recited in claim 28, wherein the step of running an instance of the application for each one of the plurality of network interfaces and the step of designating a separate one of said plurality of CPU's to each instance is performed automatically by an operating system.

39. (Currently amended) A computer system configured to provide services of an application comprising:

a plurality of network interfaces;

a plurality of CPU's, wherein each separate CPU has bound to it ~~a separate~~ only one of said plurality of network interfaces,

in an application layer, a separate running instance of the application for each one of the plurality of network interfaces, wherein a separate one of said plurality of CPU's is designated only to ~~each~~ a respective one such separate instance;

a processing queue for each of the plurality of CPUs wherein a separate one of the processing queues is assigned to each one of the plurality of CPUs and wherein the processing queue assigned to a particular CPU is configured to provide single threaded processing of data related to an instance of the application to which the CPU is bound, and

whereby each network interface is handled solely by the CPU to which ~~that~~ the said only one network interface is bound.

40. (Currently amended) ~~A~~ The computer system as in claim 39, wherein: a separate network address is assigned to each one of the plurality of network interfaces.

41. (Currently amended) ~~A~~ The computer system as in claim ~~40~~ 39, wherein each separate network address is an Internet Protocol (IP) address.

42. (Currently amended) ~~A~~ The computer system as in claim ~~40~~ 39, wherein said running instance of the application for each one of the plurality of network interfaces includes a listener that listens for the network address that is assigned to that network interface.

43. Canceled

44. (Currently amended) ~~A~~ The computer system as in claim ~~43~~ 39, wherein each processing queue is a sequential queue (s-queue).

45. (Currently amended) ~~A~~ The computer system as in claim ~~43~~ 39, wherein each single threaded processing is configured to be uninterrupted while processing the data related to an instance of the application.

46. (Currently amended) ~~A~~ The computer system as in claim 44, further configured to:

receive data packets;

process each data packet to determine a particular one of the processing queues corresponding to connection classifier information in the data packet; and

route the data packet to the determined processing queue.

47. (Currently amended) The computer system as in clam 46, further configured to:

process the data packet by the determined processing queue.

48. (Currently amended) The computer system as in claim 47, further configured to:

if the determined processing queue is busy, wait before processing the data packet by the determined processing queue.

49. (Currently amended) ~~A~~ The computer system as in claim 39, wherein the computer system is configured such that running instances of the applications network interfaces and designating a separate one of said plurality of CPU's to each instance is performed automatically by an operating system.

50. (Currently amended) A computer system comprising:

a plurality of instances of an application;

a plurality of CPU's, each CPU configured to process a separate one of said plurality of instances with each CPU having its own processing queue configured to provide single threaded processing of data related to an instance of the application;

a plurality of network interfaces for a plurality of network connections to said computer system;

an operating system, wherein said operating system is configured to:

automatically designate a separate CPU for processing ~~each separate~~ one of said separate instances of said application; and

automatically designate each of the plurality of network interfaces to only one of the plurality of CPU's, thereby assigning each one of the network interfaces to ~~an~~ only the separate instance of said application designated to the CPU to which a respective network interface is designated.